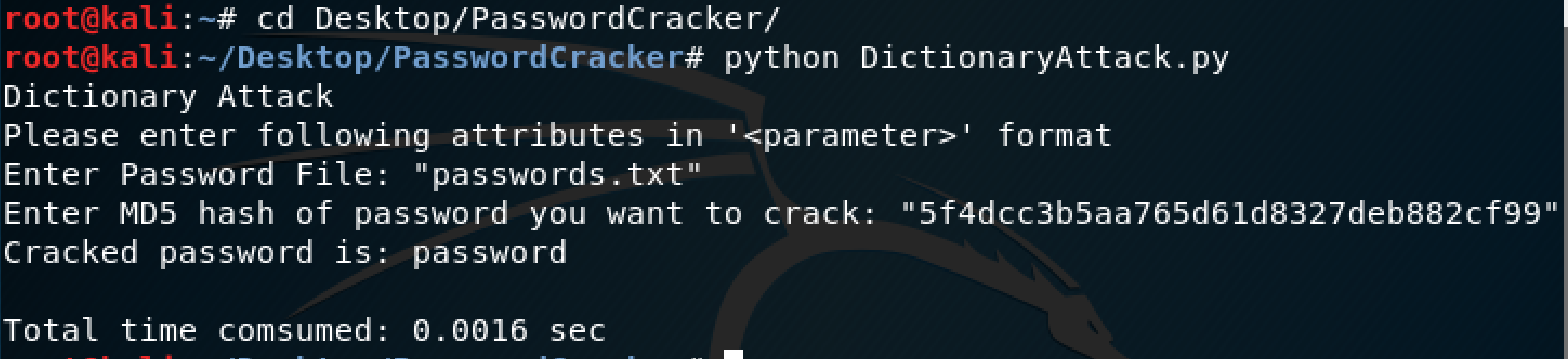
# Project 1: Password Cracking

## Dictionary Attack:

The main aim of this attack is to use a file that has well-known weak passwords. So, we traverse through the file, calculate the md5 hash for each password and compare that with a provided md5 hash. Use following command to run the python script

**#python DictionaryAttack.py**

Please provide the asked elements in a string format i.e. enclosed in “ ” or ‘ ’ if you are running from CLI. Make sure that your password file and python file are in the same directory. If it is in different directories, please provide the absolute path enclosed in “ “ or ‘ ’.



## Brute Force Attack

In this attack I am generating a password from a given set of printable string that may or may not be allowed for passwords on certain places. I am calculating the md5 hash for generated password and then comparing with the provided hash. My code for password generation uses a big string that consists of all the allowable characters. It generates password including 0-9, a-z, A-Z, special characters among the following.

! " # $ % & ' ( ) \* + , - . / : ; < = > ? @ [ \ ] ^ \_ ` { | } ~

For 1 character password, it traverses through whole list. For 2 character passwords, it will follow 00, 10, 20,…..,a0, b0,….,A0, B0,… ,!0,…,~0,01,11…~1 and so on sequence. Maximum length of password should be less than 7 and beware that more length increase the compute time exponentially.

To execute the script, use following command.

**#python BruteforceAttack.py**

Please provide the hash enclosed in “ ” or ‘ ’

